

● 嵌入式系统设计与开发系列丛书

Google Android 开发技术

Google Android Development Technology

张勇 夏家莉 陈滨 编著
蔡鹏 黄坚 张志兵



西安电子科技大学出版社
<http://www.xduph.com>

嵌入式系统设计与开发系列丛书

Google Android 开发技术

Google Android Development Technology

张 勇 夏家莉 陈 滨
蔡 鹏 黄 坚 张志兵 编著

西安电子科技大学出版社

内 容 简 介

Android 系统是安装在智能移动设备上的嵌入式操作系统,包括 Linux 内核、系统库与 Java 运行时、应用程序框架和应用程序层等四部分,具有公开源代码和免费使用的特点,是目前最受欢迎的嵌入式操作系统之一。本书介绍 Android 系统的应用程序设计方法,全书分为 8 章,内容包括 Android 系统概述、Java 语言、Android 应用程序框架、Android 单用户界面程序设计、Android 多用户界面程序设计、Android 数据访问技术、Android 图形与动画以及 Android 多媒体技术。本书的特色在于原理讲解透彻,实例丰富且有代表性。

本书是作者近几年来从事嵌入式教学和研究成果的结晶,重点阐述了 Android 应用程序设计的理论与方法,适合作为 Android 程序设计的入门书,也可作为高等院校或培训机构进行 Android 开发教学的教材和参考书。

图书在版编目(CIP)数据

Google Android 开发技术/张勇编著. —西安:西安电子科技大学出版社,2011.10

(嵌入式系统设计与开发系列丛书)

ISBN 978-7-5606-2667-3

I. ① G… II. ① 张… III. ① 移动终端—应用程序—程序设计 IV. ① TN929.23

中国版本图书馆 CIP 数据核字(2011)第 176128 号

责任编辑 李惠萍

加工编辑 李惠萍 李文娟

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfxb001@163.com

经 销 新华书店

印刷单位 陕西华沐印刷科技有限责任公司

版 次 2011 年 10 月第 1 版 2011 年 10 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 21

字 数 500 千字

印 数 1~3 000 册

定 价 38.00 元

ISBN 978-7-5606-2667-3/TN·0625

XDUP 2959001-1

*** 如有印装问题可调换 ***

前 言

智能手机的普及和移动计算技术的发展,促进了嵌入式操作系统的快速发展。一般地,某种移动设备被称为“智能的”,就是指它安装了某种嵌入式操作系统。嵌入式操作系统与通用计算机系统都是用来管理系统软、硬件资源的,它们的主要区别在于嵌入式操作系统具有实时性强和体积小的特点。目前,流行的嵌入式操作系统有 Windows CE、嵌入式 Linux、VxWorks、 μ C/OS-III等。其中,嵌入式 Linux 是免费使用且开源代码的, μ C/OS-III是公开源代码的。

Google Android 系统是基于 Linux 内核的嵌入式操作系统,严格地说是借用了其硬件驱动和线程调度功能,增加了与用户界面相关的应用程序设计框架和系统库。有些专家称 Android 不过是类似于“QT”的用户界面程序。的确,Android 在用户界面方面具有强大的优势,其特色在于优化了图形显示技术并专门设计了图标。由于从程序员的角度出发,Android 系统向应用程序提供了完备的系统调用、进程管理与进程通信以及应用程序开发接口等,因此,Android 系统被普遍认可为嵌入式操作系统,并且在全球范围内得到用户和程序员的青睐。它的最大优势在于公开了源代码且可免费使用。

本书阐述了 Android 应用程序开发的原理与技术。全书共分为 8 章。

第一章“Android 系统概述”,介绍了 Android 系统的发展历程与系统结构,并详细讲述了 Android 开发平台的建设过程,分别介绍了基于 Windows 和 Ubuntu 系统的开发平台。

第二章“Java 语言”,介绍了 Java 语言的语法与数据结构,深入讲解了 Java 类、内部类、事件方法等的概念。由于 Android 应用程序采用 Java 语言编写,因此这一章的内容可使那些没有接触过 Java 语言的读者快速入门。

第三章“Android 应用程序框架”,介绍了 Hello World 工程及其结构,详细讲述了 Hello World 工程的工作原理及应用程序框架的基本组成,并分析了 Activity(活动界面)的生命周期。

第四章“Android 单用户界面程序设计”,讲述了 Activity 的概念和使用方法,详细讲解了应用程序布局方法和 Android 系统常用控件的种类及其使用方法。通过“计算器”工程深入分析了单用户界面程序设计的特点。

第五章“Android 多用户界面程序设计”,介绍了 Intent 的概念和不同界面间的数据通信方法。与单用户界面程序相比,多用户界面程序需要进行界面间的数据通信,包括对话框与 Activity 之间、菜单与 Activity 之间以及两个 Activity 之间的数据通信,借助于内部类或 Intent 对象可实现这些通信方法。

第六章“Android 数据访问技术”,介绍了 Android 系统的四种数据存储与访问方式,即 SharedPreferences 文件访问、流文件操作、SQLite 关系数据库和 Content Provider(内容提供者)等,并通过实例对比了这四种方式的异同点。其中,内容提供者可实现不同应用程序间的数据共享和通信。

第七章“Android 图形与动画”，详细讲述了借助 View 类和 SurfaceView 类进行图形绘制和动画设计的方法，介绍了图形绘制的应用程序框架，阐述了三种动画方式，即定时器动画、渐变动画和帧切换动画。

第八章“Android 多媒体技术”，介绍了借助 MediaPlayer 类播放音频文件和视频文件的方法，并介绍了 Service(服务)的程序设计方法。

本书若用于大学课内教学，应讲述全部内容，理论学时宜为 32 学时，课内实验学时不少于 16 学时。本书的特色在于 Android 应用程序设计原理讲解透彻，实例丰富且具有代表性。本书的实例代码可以在西安电子科技大学出版社网站(<http://www.xduph.com>)下载，或通过 E-mail: zhangyong@jxufe.edu.cn 向作者索取。

如果自学本书，除了掌握本书全部内容外，还应该通过本书的学习，学会查阅 Android 开发者手册，这个手册位于 Android SDK 安装目录的 docs 子目录下。

本书中的一些图像和音、视频素材均来自于互联网，其版权归相关公司所有，书中引用的这些素材仅用于教学和研究。同时，作者保留其他内容的所有权利。

最后，作者感谢北京博创科技、广州天嵌科技和深圳英蓓特公司提供了测试平台，并审校了初稿，他们始终为作者撰写“嵌入式系统设计与开发系列”丛书提供技术支持。同时，感谢这套丛书的热心读者反馈的宝贵意见，感谢西安电子科技大学出版社的李惠萍编辑为本书出版所做的细致工作。由于作者水平有限，书中难免有纰漏之处，敬请同行专家和读者批评指正。

张 勇

2011 年于江西财经大学枫林园

目 录

第 1 章 Android 系统概述.....1	
1.1 Android 操作系统的概念.....2	
1.2 Android 系统结构.....4	
1.3 Android 开发平台.....7	
1.3.1 基于 Windows 的 Android 开发平台.....7	
1.3.2 基于 Ubuntu 的 Android 开发平台.....18	
1.4 小结.....25	
第 2 章 Java 语言.....27	
2.1 Java 程序语法与控制.....27	
2.1.1 顺序方式.....27	
2.1.2 分支方式.....31	
2.1.3 循环方式.....36	
2.1.4 异常处理.....39	
2.2 Java 基本数据类型.....41	
2.2.1 数值.....41	
2.2.2 字符.....43	
2.2.3 字符串.....44	
2.2.4 布尔数.....45	
2.2.5 数组.....46	
2.3 Java 类.....47	
2.3.1 类与对象.....48	
2.3.2 继承与多态.....50	
2.3.3 接口.....54	
2.4 Java 文件操作.....56	
2.5 在命令行窗口中运行 Java 程序.....60	
2.6 Java 图形界面.....62	
2.6.1 事件响应方法.....62	
2.6.2 内部类.....64	
2.6.3 匿名内部类.....66	
2.7 小结.....67	

第 3 章 Android 应用程序框架..... 68	
3.1 Hello World 工程..... 68	
3.2 Hello World 工程工作原理..... 72	
3.3 应用程序框架..... 77	
3.3.1 应用程序框架基本组成..... 77	
3.3.2 Android 配置文件 AndroidManifest.xml..... 78	
3.3.3 Android 资源文件..... 81	
3.3.4 Android 源程序文件..... 86	
3.4 Activity 生命周期..... 89	
3.5 小结..... 93	

第 4 章 Android 单用户界面 程序设计..... 95	
4.1 Activity 的概念..... 95	
4.2 布局与控件..... 96	
4.2.1 布局软件 DroidDraw..... 96	
4.2.2 控件事件响应方法..... 105	
4.2.3 Android 常用控件..... 112	
4.2.4 线性布局..... 115	
4.2.5 相对布局..... 118	
4.2.6 框架布局..... 120	
4.2.7 表格布局..... 121	
4.2.8 绝对布局..... 125	
4.3 计算器工程..... 132	
4.4 小结..... 144	

第 5 章 Android 多用户界面 程序设计..... 145	
5.1 Intent 的概念..... 145	
5.2 对话框..... 148	
5.2.1 AlertDialog 对话框..... 148	
5.2.2 自定义对话框..... 154	
5.2.3 Dialog 类..... 159	

5.2.4 ProgressDialog 对话框.....	166	6.5 小结.....	263
5.3 菜单.....	170	第 7 章 Android 图形与动画	264
5.3.1 XML 布局菜单.....	171	7.1 绘图.....	264
5.3.2 动态菜单	175	7.1.1 View 类绘图程序框架	265
5.3.3 上下文菜单	177	7.1.2 SurfaceView 类绘图程序框架	275
5.4 多用户界面设计	182	7.1.3 基本图形与字符串.....	279
5.4.1 简单多用户界面显示	182	7.2 动画.....	280
5.4.2 多用户界面的数据传递	192	7.2.1 定时器动画.....	280
5.4.3 活动界面间的双向数据通信	195	7.2.2 渐变动画.....	292
5.5 小结	208	7.2.3 帧切换动画.....	297
第 6 章 Android 数据访问技术	209	7.3 小结.....	301
6.1 SharedPreferences 文件访问.....	209	第 8 章 Android 多媒体技术	303
6.2 流文件操作	223	8.1 音频文件播放.....	303
6.3 SQLite 关系数据库.....	237	8.2 服务(Service).....	313
6.3.1 SQLite 数据库访问方法.....	237	8.3 视频文件播放.....	319
6.3.2 SQLiteOpenHelper 类	244	8.4 小结.....	329
6.4 内容提供者(Content Provider)	251		



第 1 章 Android 系统概述

Android 音译为“安卓”。Android 系统是安装在移动设备(例如智能手机、个人数字助理(PDA)、MP5 播放器、手持终端、平板电脑、上网本、电子书等)上的操作系统软件,用于管理和调度移动设备的软、硬件资源。其作用相当于个人电脑(PC)上安装的微软 Windows 操作系统,与安装在智能手机上的 Windows Mobile Phone(Windows CE)操作系统相似。Android 系统与桌面 Windows 系统、Windows CE 操作系统的异同点列于表 1-1,从表 1-1 中所列的特点,我们可以对 Android 系统有一个全面直观的认识。

表 1-1 Android 系统与桌面 Windows 系统、Windows CE 操作系统的异同点

比 较 项 目	Android 系统	桌面 Windows 系统	Windows CE 系统
用户界面	有	有	有
标准内存	256 MB	2 GB	256 MB
系统大小	<64 MB	约 4 GB	<64 MB
内核驻留	Flash 芯片	硬盘	Flash 芯片
实时性	强	弱	强
应用设备	嵌入式移动设备	个人电脑	嵌入式移动设备
功耗	低	高	低
应用软件体积	小	大	小
应用软件兼容性	极强	强	强
系统源代码	公开	不公开	核心不公开
内核独立性	基于 Linux	独立	独立
输入设备	触摸屏为主	键盘、鼠标为主	触摸屏为主
输出分辨率	相对低	相对高	相对低
英文直译	科幻机器人	视窗	便携式电子窗口
开发商	Google(谷歌)为主	微软(Microsoft)	微软

Android 系统由 Andy Rubin 首创,最初的目的是设计一种新的开放性智能手机操作系统。2003 年美国就已经有大量的移动设备使用嵌入式操作系统,新研发的操作系统想进入市场并被用户认可是件很困难的事情。然而当时的大部分嵌入式操作系统都不是开源的,维护十分困难,Rubin 等人因此提出这种开源的智能手机操作系统,希望借此挤进激烈竞争、商机无限的嵌入式操作系统市场中,其赢利主要靠安装、维护和提供专业特色应用软件等技术服务。现在看来,Rubin 的做法成功了。2005 年 8 月 Google 收购 Android 加速了该开



源嵌入式操作系统的发展, 2007 年以 Google 为首组建了全球性的开放手机联盟(Open Handset Alliance), 中国电信、中国移动和中国联通也是其中的成员, 在全球范围内推动基于 Android 操作系统的手机开发计划。2008 年 10 月宏达电(HTC)公司推出了第一款 Android 系统的手机, 命名为 HTC Dream(G1), 如图 1-1(a)所示, 这是一款被市场证实成功的手机。随后, 几乎在全球形成了研究 Android 操作系统的热潮。Android 操作系统的功能和版本逐年提高, 目前已经是第 2.3.3 版, 内部研发版本则更高。图 1-1(b)是基于 Android 2.3.3 版本的 Flyer 智能手机。2011 年初, Android 已经成为嵌入式操作系统领域最受欢迎的智能操作系统。

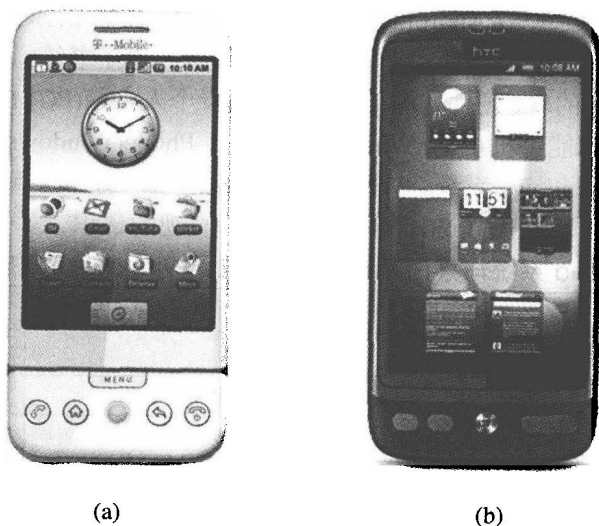


图 1-1 HTC Dream (G1) 和 HTC Flyer

Android 操作系统是开放源代码的, 并且拥有全球最多的研究人员和用户群, 源文件中的大量 Bug(问题)会被及时发现而纠正, 因此, Android 系统版本更新频繁。但是, 基于 Android 系统的应用程序开发技术在各个版本中的方法完全相同, 这些正是本书要介绍的内容。(注: 本书内容基于 2011 年初结稿时的最新 Android 系统版本 2.3.3)。

1.1 Android 操作系统的概念

Android 操作系统是基于 Linux 内核的嵌入式操作系统, 其底层(称为第一层)为 Linux 操作系统及其驱动, 该层源代码是用 C 语言编写的。底层上面建构了系统库和 Java 运行时(即 Java 程序运行支持软件包或 Java 虚拟机, “运行时”是由“Runtime”意译而来, 在很多书上均采用这一译法), 称为第二层, 这一层是使用 C/C++代码写成的。第三层为应用程序框架层, 为用户开发 Android 程序直接提供 API(应用程序接口)函数, 这一层是用 Java 代码实现的。第四层为用户应用程序层。由于 Android 操作系统内置了许多用户应用程序, 因此有些专家认为应用程序层可以划分到 Android 操作系统中。当然, 用户自己编写的应用程序也属于这一层。这一层的应用程序使用 Java 语言来设计。Android 系统结构如图 1-2 所示。



Android 系统结构将在第 1.2 节中详细介绍。从图 1-2 中可以看出, Android 操作系统采用分层结构, 且整个系统建立在 Linux 操作系统内核基础上, 借助 Linux 内核硬件驱动进行硬件资源的管理。因此, Android 系统没有独立的硬件底层驱动部分。事实上, Android 系统的软件调度也借助了 Linux 内核进程调度实现, 即两个显著的操作系统特征在 Android 系统下没有得到体现, 从严格意义上讲, Android 系统应该隶属于应用软件系统的范畴。而与 Android 系统竞争市场的 Windows CE 操作系统则完全不同, 它包括完整的内核层、驱动层和应用程序层, 是真正意义上的嵌入式操作系统。如果从应用程序开发者的角度出发, 而不去考虑图 1-2 所示的 Android 系统结构, 由于 Android 系统封装了各层间的通信和服务调用, 向应用程序开发者提供了完备的系统调用(包括驱动程序开发)、进程管理与进程间通信和应用程序开发接口等, 因此, 从这个意义上说, Android 系统属于操作系统的范畴。现在, Android 系统研发者和应用程序开发者都普遍认可 Android 系统属于嵌入式操作系统, 从概念上将它与 Windows CE 等嵌入式操作系统等同。

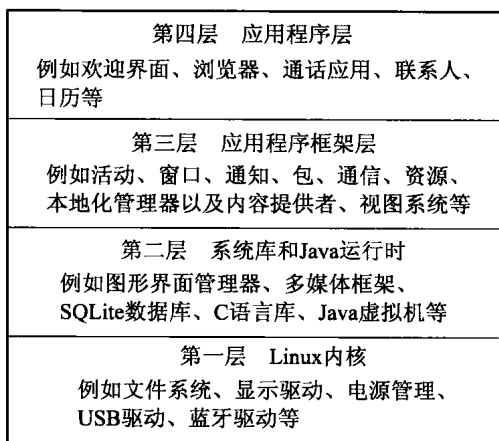


图 1-2 Android 系统结构

Android 系统相对于其他嵌入式操作系统而言, 具有两个明显的优点, 即开放源代码和网络功能强大。前面提到了 Android 系统最初开放源代码的原因, 从 2003 年到今天仍然保持着这一独特的优势, 除了嵌入式操作系统领域市场竞争激烈外, Android 使用 Linux 作为其底层平台是其开源的另一个重要原因。Google 本身是互联网公司, 其下的所有产品都是基于互联网模式发展的, Google 收购的 Android 系统也不会例外。伴随着 Android 系统的诞生和版本升级, Android 系统的网络功能越来越强大, 这使得基于 Android 系统编写网络程序比基于其他任何嵌入式操作系统都更加容易。可以说, 一部 Android 手机就是一部互联网终端, 网上购物、新闻、旅游、导航、智能家居等应用的确给用户带来了极大的生活便利。

Android 系统使用 Java 语言编写应用程序, 从一定意义上可以说 Android 系统推动了 Java 语言的广泛应用。Java 语言属于面向对象的高级语言, Java 语言程序必须借助于 Java 虚拟机解释执行, 它比其他高级语言的可移植性都强。在 Android 模拟器上运行成功的应用程序, 一定能够成功地部署和运行在 Android 系统终端机上, 这使得 Android 系统应用程序的开发变得非常方便。

Android 系统的图形界面也是它的一个亮点, 严格地说, Android 系统不是基于可视化窗口的, 而是直接基于图形的。也就是说, Android 系统界面是由一幅幅图画组合在一起的, 因此, Android 系统界面比较“炫”! 相比于 Windows CE 的视窗而言, 其更人性化一些。Android 系统界面美观是其受到用户欢迎的最重要的原因, 尽管如此, Google 对现有 Android 系统界面仍然不很满意, 据说新版本的 Android 系统在用户界面上还会有较大的创新。

目前最新的 Android 系统版本号为 2.3, 研发代号为 Gingerbread(姜饼), 其主要的功能和特点如下:



- (1) 系统源代码公开, 通过 SDK(Software Development Kit)和模拟器 AVD(Android Virtual Device)可以在 PC 机上体验 Android 2.3 的功能;
- (2) 支持无线通信和 3G 网络;
- (3) 用户界面可定制为个性化界面;
- (4) 支持互联网功能、后台下载和在线文件夹(即网络文件夹), 采用 Google 搜索服务;
- (5) 支持多摄像头、陀螺仪等多种传感器, 这样一部 Android 手机可以集成前、后两个摄像头;
- (6) 具有强大的多媒体功能, 支持混响音效(例如低音、环绕和虚拟化效果等);
- (7) 具有强大的图像管理功能, 支持 JPG、GIF、PNG 和 BMP 等格式图像文件;
- (8) 支持 Web 浏览器;
- (9) 支持 WiFi、蓝牙和近距离通讯(Near Field Communications, NFC)等;
- (10) 具有地图、定位和 GPS 功能等。

Android 系统是纯粹的商业性操作系统, 在 GPL(General Public License)协议条件下源代码公开且免费使用, 这意味着当用户免费使用 Android 系统开发软件产品时, 其衍生的软件产品也必须是开源和免费的。需要指出的是, Android 虚拟机不是开源的。

1.2 Android 系统结构

图 1-2 为 Android 系统结构图, 由于直接翻译 Android 系统组件的术语并不准确, 因此这里给出了经典的英文 Android 系统结构图, 如图 1-3 所示。

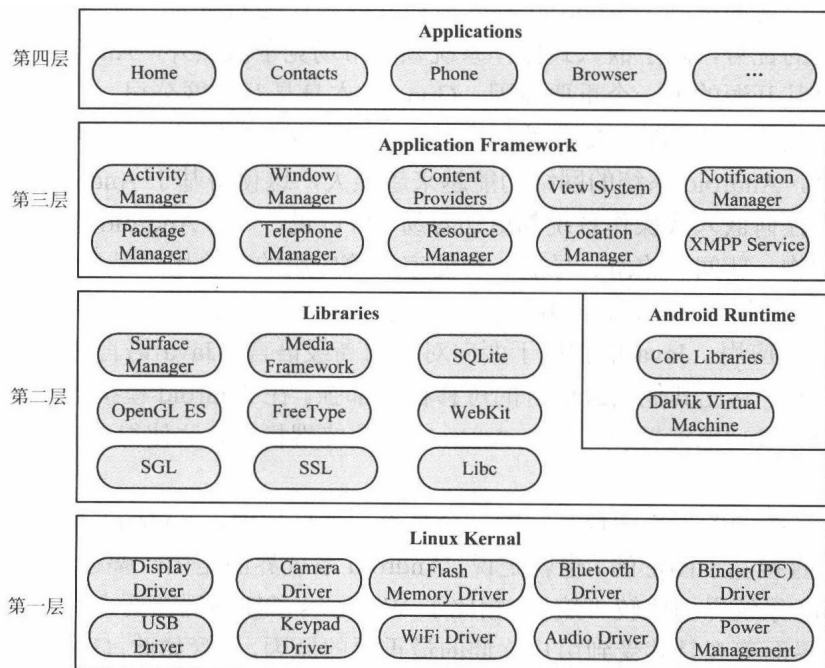


图 1-3 Android 系统结构图

由图 1-3 可知, Android 系统是基于 Linux 内核的操作系统, 习惯上把 Linux 内核(Linux



Kernel)层称为其第一层。Linux 是免费和开源码的实时抢先式多任务操作系统, Linux 内核协助 Android 系统完成进程调度、进程间通信、内存管理、虚文件系统管理、系统安全管理和设备驱动等功能。图 1-3 中仅列举了 Linux 内核实现的 10 种设备驱动功能, 即显示驱动、摄像驱动、Flash 存储驱动、蓝牙驱动、Binder IPC 驱动(用于进程间通信管理)、USB 驱动、键盘驱动、WiFi 驱动、音频驱动、功耗管理。事实上, Linux 内核还协助 Android 完成强大的网络管理和驱动等, 被视为 Android 系统的一个硬件抽象层。正因为 Android 基于 Linux 内核, 很多专家指出, 要深入学习 Android 必须加强 Linux 系统的学习。但如果重点放在 Android 应用程序设计上, 即使不懂 Linux, 对学习 Android 程序开发也影响不大。由于 Android 系统是架构在 Linux 系统之上的, 因此 Linux 系统不支持的处理器, Android 系统也同样无法支持, 即 Android 系统只能移植到 Linux 系统可以运行的处理器上。好在 Linux 系统支持大多数流行的处理器, 例如 x86 结构、ARM、MIPS 等。Linux 系统的这种广泛的可移植性决定了 Android 系统具有广泛的可移植性。

第二层是 Android 系统库(Libraries)和 Android 运行时(Runtime), 这一层是使用 C/C++ 代码编写的。Android 系统库包含了大量的类, 通过第三层(即应用程序框架层)被应用程序开发者调用, 应用程序开发者使用的大量 API(应用程序接口)函数来自于这些类。API 函数越丰富, 则用户开发应用程序的工作量越小。随着 Android 系统版本的升级, API 函数级别也随之上升, 如表 1-2 所示。

表 1-2 Android 系统与 API 级别的关系

Android 系统版本号	代号	API 级别(Level)
1.1	—	2
1.5	Cupcake	3
1.6	Donut	4
2.0	Eclair	5
2.0.1	Eclair	6
2.1	Eclair	7
2.2	Froyo	8
2.3.1	Gingerbread	9
2.3.3	Gingerbread	10
2.4	Gingerbread	未公开
3.0	Honeycomb	11
3.1	Icecream	未公开

表 1-2 显示的最新的 API 级别为 11。从表 1-2 中可以看到 Android 系统每个版本的研发代号都是一种食品的名称, 依次译为“杯形蛋糕”、“炸面圈”、“指形小饼”、“冻酸奶”、“姜饼”、“蜂巢”和“冰淇淋”, 估计发起人是一位美食家, 可以预见下一个版本一定也是一种流行的受人喜爱的小食品名称, 这也正是 Android 系统研发的用意所在, 希望能受到系统研发者和应用程序开发者们的喜爱。

图 1-3 中第二层 Android 系统库给出了 9 个组件, 即 Surface Manager(界面管理器)、Media Framework(多媒体框架)、SQLite、OpenGL ES、FreeType、WebKit、SGL、SSL 和 Libc。这



9 个组件都十分复杂，下面简要说明一下各个组件的作用。

Surface Manager(界面管理器)负责显示相关的操作，Android 系统界面是基于图形系统的，这种图形系统采用客户端/服务器的方式进行工作。“客户端”就是用户的应用程序界面，而“服务器”负责与这个应用程序界面相关的数据管理，由一个称为 **Surface Flinger** 的组件管理，“服务器”与“客户端”的通信需要借助于 **Binder** 类。界面管理器就是要管理这种工作方式，以实现二维或三维图形的显示。

Media Framework(多媒体框架)是一个非常实用的类库，其中封装了大量处理多媒体数据的 API 函数，支持流行的绝大部分多媒体格式，使得应用程序开发者开发多媒体软件时非常轻松，只需要调用多媒体框架中的 API 函数就可以了。

SQLite 是 Android 集成的关系型数据库，也是开源代码的嵌入式数据库，支持 ANSI SQL92 标准的大部分 SQL(结构化查询语言)语句，它速度快、体积小(约 250 KB)，最大能支持的数据库文件大小为 4 TB。

OpenGL ES 是 Android 系统中二维和三维图形处理与加速的 API 函数集。

SGL 是 **Skia Graphics Library** 的首字母简写，是 Android 用来处理二维图形的向量图形引擎。所谓的“引擎”术语是从汽车术语借用过来的，在计算机软件中的引擎是指软件处理中的最核心部分，就像汽车发动机是汽车的核心一样，一个软件功能的升级主要取决于其“引擎”部分的升级。**SGL** 就是 Android 系统的二维图形引擎。

FreeType 是 Android 系统使用的字体引擎，**FreeType** 是开源码和免费的，其优点在于提供简单、统一的 API 函数访问多种字体格式文件，例如位图字体和矢量字体，这使得 Android 系统处理字体时非常方便。

SSL 是 **Secure Sockets Layer** 的首字母简称，即安全套接层，这说明 Android 系统支持 **SSL**，即安全套接层协议。**SSL** 对发送的网络数据进行加密，防止数据在传送至合法目的地网络终端的过程中被非法用户使用或修改。这在电子商务和网上银行的数据交换中尤为重要。**SSL** 通过对合法用户的认证和数据的加密，确保用户的信息安全，Android 系统自诞生以来，就支持 **SSL** 协议。事实上，所有的嵌入式操作系统浏览器都支持 **SSL** 协议。

WebKit 是开源码的网络浏览器引擎，这个引擎稳定性好、兼容性强、效率高，不仅 Android 系统浏览器基于 **WebKit** 引擎，苹果的 iPhone 浏览器也基于这个引擎。

libc 是通用的 C 语言库，供 Android 系统库调用。

Android 运行时(Runtime)包括核心库(Core Libraries)和 **Dalvik** 虚拟机(Virtual Machine)。核心库集成了绝大多数 Java 语言核心库的功能，供 Java 语言程序运行时调用。**Dalvik** 虚拟机解释并运行格式为 **dex** 的 Java 程序，术语 **dex** 是 **Dalvik eXecutable** 的缩写，常规的 Java 语言程序(class 字节文件)通过 Android 系统内置的 **dx** 工具转化为 **dex** 格式，这种格式被优化为代码内存占用最小的文件格式，因此，Android 的可执行文件扩展名为 **.dex**。每个 Android 应用程序启动后都对对应着一个进程，该进程属于它自己的 **Dalvik** 虚拟机实例。**Dalvik** 虚拟机可以同时高效地运行多个虚拟机实例，从而实现多任务处理。

第三层为 Android 应用程序框架，进行 Android 应用程序开发必须熟练掌握其 4 种基本组件，即活动(Activity)、服务(Service)、广播接收器、内容提供者(Content Provider)的使用技术。应用程序框架层为开发 Android 应用程序提供了各种 API 函数，这些函数属于不同



的类。图 1-3 中列出了 10 类组件,即活动管理器(Activite Manager)、窗口管理器(Window Manager)、内容提供者(Content Providers)、视图系统(View System)、通知管理器(Notification Manager)、包管理器(Package Manager)、电话管理器(Telephone Manager)、资源管理器(Resource Manager)、地址管理器(Location Manager)、XMPP 服务(XMPP Service)。其中, XMPP 是 Google Talk 的通信协议,而 Google Talk 简称 GTalk,是 Google 的即时通讯方式,也就是通常所说的文字或语音聊天。此外, GTalk 还支持 E-mail 功能。第三层是进行应用程序开发的基础,也是本书涉及的主要内容,这一层的组件将在以后的章节中详细介绍。

第四层为用户应用程序层,这一层的软件包括欢迎界面(Home)、联系人(Contacts)、电话(Phone)、浏览器(Browser)等用户直接使用的程序,当然也包括用户自己开发的应用程序。本书将详细介绍该层应用程序的设计方法。

1.3 Android 开发平台

通过 1.1 节和 1.2 节的介绍,我们基本上对 Android 系统有了一个全面的认识。本节介绍如何组建基于 Android 系统的应用程序开发环境,这里介绍基于 Windows 操作系统和基于 Ubuntu 操作系统两种开发平台建设方式,这两种开发平台都是 Android 程序员常用的工作模式。需要指出的是,如果用户使用 Ubuntu 操作系统组建 Android 开发平台,则所有软件都是免费的,即整个运行和开发环境(包括 Ubuntu 系统在内)都是免费的。

1.3.1 基于 Windows 的 Android 开发平台

组建 Android 开发平台需要的软件有四种,即 JDK(Java Development Kit)、Eclipse、ADT(Android Development Tools)插件、Android SDK(Software Development Kit)。令人欣慰的是,所有这些软件都可以免费获得和使用。

为了使组建平台的过程条理性比较强,下面按安装步骤依次介绍,每一步用“S+序号”表示,例如第一步用 S1 表示。笔者使用的计算机操作系统为 32 位 Windows XP 专业版 SP3,但是下面介绍的步骤适用于更高版本的 Windows 操作系统,即这里介绍的安装步骤具有通用性。建议读者使用最新的 Windows 操作系统。

S1. 安装 JDK 软件

如果要在 Windows 操作系统下运行和开发 Java 程序,必需安装 JDK(Java Development Kit),JDK 包含了 JRE(Java Runtime Environment)。如果只想在 Windows 系统下运行 Java 应用程序,则只需要安装 JRE。

登录 <http://www.oracle.com/technetwork/java/javase/downloads/index.html> 下载 Java SE Development Kit 6u24 (JDK),文件名为“jdk-6u24-windows-i586.exe”,大小约为 76.58 MB。进入到该网页的超链接“Installation Instructions”(安装向导),可以阅读详细的安装方法。本书所用的 JDK 软件版本为截止 2011 年 4 月中旬时的最新版本。JDK 软件的更新比较快,建议读者安装最新版本的 JDK 软件。

双击下载后的程序图标“jdk-6u24-windows-i586.exe”,即进入安装 JDK 的过程。笔者采用了缺省的安装目录“C:\Program Files\Java\jdk1.6.0_24\”,安装过程中将提示安装 JRE,



笔者采用了缺省目录“C:\Program Files\Java\jre6”。一般地，计算机将在 2 分钟内完成安装，然后进入到“所有程序|附件|命令提示符”窗口，输入“java -version”，将显示如图 1-4 所示结果，表明安装 JDK 软件成功。

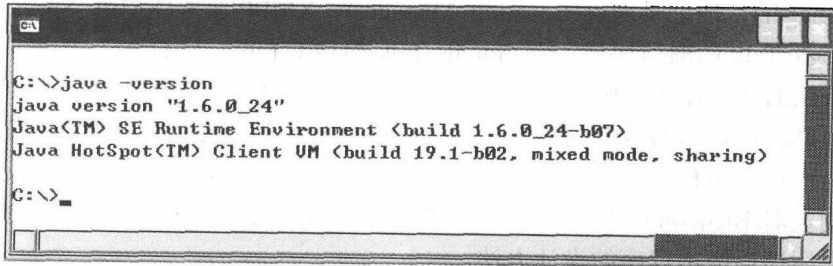


图 1-4 显示 JDK 版本号窗口

S2. 安装 Eclipse 软件

登录 <http://www.eclipse.org/> 下载 Eclipse 软件，网站上的各个 Eclipse IDE(集成开发环境)均可以使用，这里推荐使用 Eclipse IDE for Java EE Developers 或 Eclipse Classic 3.6.2。笔者使用 Eclipse IDE for Java EE Developers，文件名为“eclipse-jee-helios-SR2-win32.zip”，大小约为 206 MB，截止到 2011 年 4 月中旬的最新版本号为 4.1。建议读者下载最新版本的 Eclipse 软件。该软件无需安装，解压后即可运行，笔者将其解压到 D 盘，如图 1-5 所示，启动文件名为“eclipse.exe”。

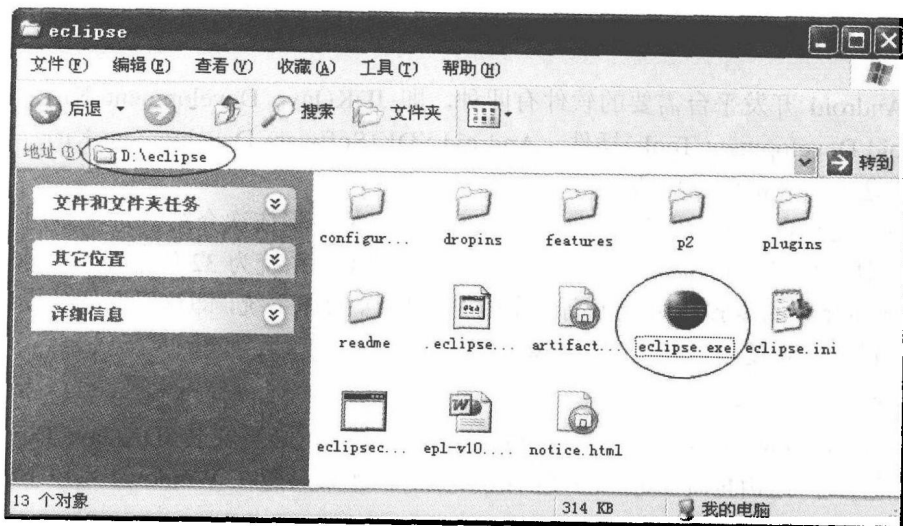


图 1-5 Eclipse 软件包

双击图 1-5 中的图标“eclipse.exe”，第一次使用时会弹出如图 1-6 所示窗口。在“Workspace”中填入一个路径，笔者使用“D:\myWorkspace”这个路径，或者使用图 1-6 右边的“Browse...”按钮选择其他已存在的路径。然后，选中“Use this as the default and do not ask again”复选框，在以后运行 Eclipse 软件时默认的工作区保存目录为“D:\myWorkspace”，将不再弹出该启动窗口。

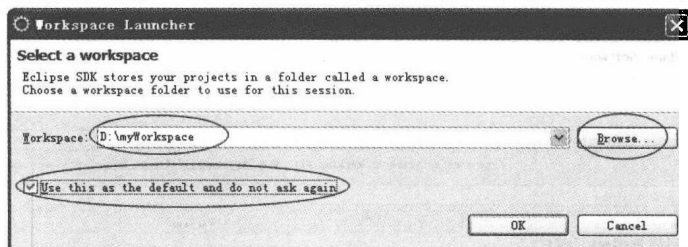


图 1-6 选择工作区目录

点击图 1-6 中的“OK”按钮进入到 Eclipse 软件主界面，如图 1-7 所示。

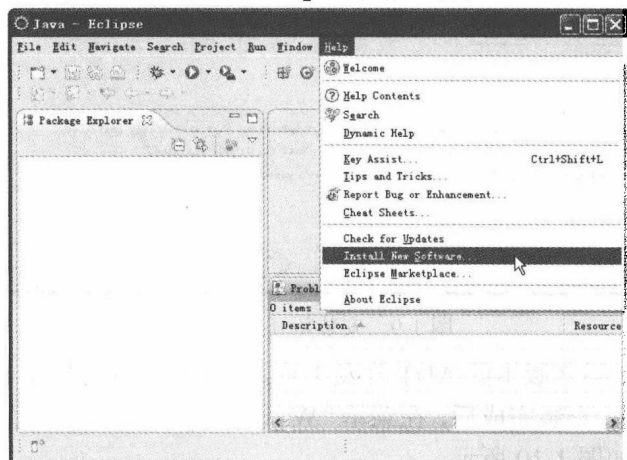


图 1-7 Eclipse 软件主界面

S3. 安装 ADT 插件

在图 1-7 中选择菜单项“Help | Install New Software...”，进入到图 1-8 所示的窗口，在“Work with”栏中输入 <http://dl-ssl.google.com/android/eclipse/>。然后，点击右边的“Add...”按钮，在弹出的“Add Repository”对话框的“Name”栏中输入“ADT”（此处可以输入其他读者想要的名称），然后点击“OK”按钮，进入到图 1-9 所示界面。

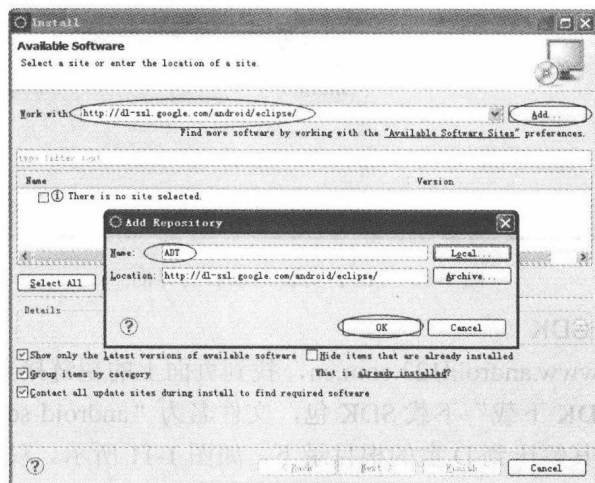


图 1-8 安装新软件

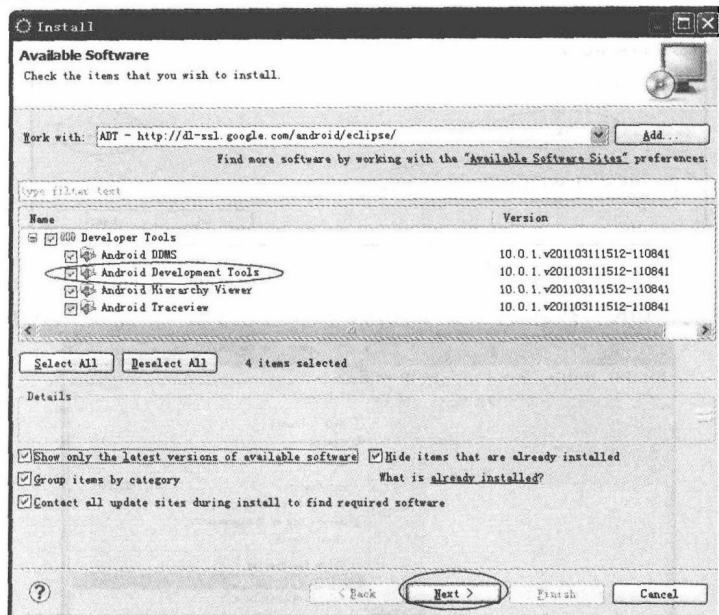


图 1-9 安装新开发工具

在图 1-9 中显示了最新版本的 ADT 开发工具，选中所有的开发工具，然后点击“Next”按钮进行 ADT 的安装。安装完成后，在菜单“Window”中多了一个子菜单项“Android SDK and AVD Manager”，如图 1-10 所示。

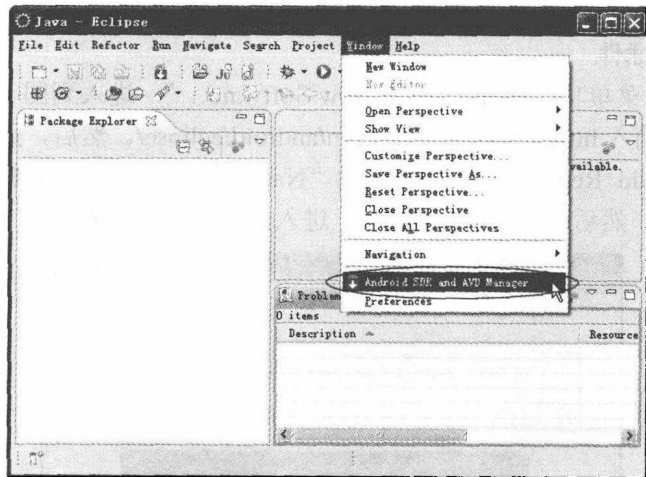


图 1-10 ADT 安装完成后的界面

S4. 安装 Android SDK

登录到网站 <http://www.android123.com.cn>，找到页面上的超链接“SDK 下载”，进入之后找到“Android 2.3 SDK 下载”下载 SDK 包，文件名为“android-sdk_r08-windows.zip”，约有 31.1 MB，笔者将其解压到 D 盘的根目录下，如图 1-11 所示。双击图 1-11 中的可执行文件“SDK Manager.exe”，联网在线安装最新的 Android SDK(截止 2011 年 4 月最新版本为 Android 3.0 Revision 1，本书中使用的版本为 2.3.3 Revision 1)，如图 1-12 所示。笔者安装了